AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A method for encrypting data in a computer in communication with a volatile memory and non-volatile storage device, comprising:

initiating a paging operation to move ~~encrypting~~ pages in the volatile memory ~~to move~~ to a swap file in the non-volatile storage device, wherein the non-volatile storage device is ~~as~~ part of a virtual addressing system;

generating codes to use to encrypt and decrypt the pages, wherein the codes are permanently lost if the computer performs a boot operation;

encrypting the pages in the volatile memory;

moving the encrypted pages from the volatile memory to the swap file; and

upon receiving a subsequent request to transfer the encrypted pages from the swap file to the volatile memory,

decrypting the encrypted pages in the swap file ~~to move back into the volatile memory;~~, and

moving the decrypted pages ~~in the swap file~~ back into the volatile memory.

2 (Canceled).

3. (Currently amended) The method of ~~claim 2~~claim 1, wherein the codes comprise a public/private key pair generated using a public key cryptography

algorithm, wherein one key of the pair is used to encrypt the pages moved to the swap file and the other key of the pair is used to decrypt the page when moving the page from the swap file to the volatile memory.

4 (Canceled).

5. (Currently amended) The method of ~~claim 2~~claim 1, wherein the codes are loaded into a non-swappable region of the volatile memory that is not moved to the swap file.

6. (Original) A method for encrypting files in a computer file system in communication with a volatile memory and a non-volatile storage device, wherein files in the file system are associated with groups, comprising:

providing, for each group, a group identifier, a list of user identifiers of users allowed to access files in the group, and a first encryption code;

receiving a second encryption code for one user identifier;

receiving an input/output (I/O) request from a requesting user identifier with respect to a target file, wherein one second encryption code has been received for the user identifier;

determining the group associated with the target file and the first encryption code for the group;

if the I/O request is a write operation, then using the determined first encryption code to encrypt the target file to write the target file to the non-volatile storage device; and

if the I/O request is a read operation to read the target file from the non-volatile storage device, then performing:

(i) determining whether the requesting user identifier is in the list of the determined group; and

(ii) if the requesting user identifier is in the list, then using the second encryption code for the user identifier to decrypt the target file.

7. (Original) The method of claim 6, further comprising:

for each group, generating a public and private encryption key pair using a public key encryption algorithm, wherein the first encryption code for the group is one of the generated public key or private key and the second encryption code is the other one of the public or private key generated for the group.

8. (Original) The method of claim 7, further comprising receiving a plurality of keys from the user, wherein each received key is used to decrypt files associated with one group identifier.

9. (Original) The method of claim 7, further comprising:

generating an index entry in a non-swappable region in the volatile memory; and

adding to the index entry the user identifier of the user that provided they key, the group identifier associated with the received key, and the received key.

10. (Original) The method of claim 9, wherein the index entry for the user identifier and group identifier is only generated if the user identifier is included in the list associated with the group identifier, and wherein the user identifier cannot perform a read access for the target file if there is no index entry for the group identifier associated with the target file and the user identifier.

11. (Original) The method of claim 9, wherein files read and decrypted from the non-volatile storage device are cached in the volatile memory, and wherein if the requested file is unencrypted in the cache, returning the

unencrypted file from the cache to the requesting user identifier if the requesting user identifier is in the list associated with the group identifier and there is one index entry for the user identifier and group identifier in the volatile memory.

12. (Currently amended) The method of ~~claim 1~~claim 6, wherein the second encryption code is accessed from a removable storage medium.

13. (Original) A method for encrypting files in a computer in communication with a volatile memory and non-volatile storage device, comprising;

generating an encryption code to encrypt a file and a decryption code to decrypt one file encrypted with the encryption code;

loading the decryption code into the volatile memory, wherein the decryption code is erased from the volatile memory when the computer reboots;

encrypting files with the encryption code to transfer from the volatile memory to the non-volatile storage device; and

decrypting files with the decryption code maintained in the volatile memory to transfer from the non-volatile storage device to the volatile memory.

14. (Original) The method of claim 13, further comprising:

generating a new encryption and decryption codes when the computer reboots, wherein the new encryption code is used to transfer files from the volatile memory to the non-volatile storage device and wherein the new decryption code is used to transfer files from the non-volatile storage device to the volatile memory as part of a read operation.

15. (Original) The method of claim 13, wherein the decryption code is loaded into a non-swappable region of the volatile memory.

16. (Original) The method of claim 13, wherein the files are transferred between the volatile memory and non-volatile storage as part of a virtual memory paging operation.

17. (Currently amended) A system for encrypting data, comprising:

a volatile memory;

a non-volatile storage device, wherein data is capable of being transferred between the volatile memory and non-volatile storage device;

means for initiating a paging operation to move ~~encrypting~~ pages in the volatile memory ~~to move~~ to a swap file in the non-volatile storage device, wherein the non-volatile storage device is ~~as~~ part of a virtual addressing system;

means for generating codes to use to encrypt and decrypt the pages, wherein the codes are permanently lost if the computer performs a boot operation;

means for encrypting the pages in the volatile memory;

means for moving the encrypted pages from the volatile memory to the swap file;

means for receiving a subsequent request to transfer the encryhpted pages from the swap file to the volatile memory;

means for decrypting the encrypted pages in the swap file ~~to move back into the volatile memory~~; and

means for moving the decrypted pages ~~in the swap file~~ back into the volatile memory.

18 (Canceled).

19. (Currently amended) The system of ~~claim 18~~claim 17, wherein the codes comprise a public/private key pair generated using a public key cryptography algorithm, wherein one key of the pair is used to encrypt the pages

moved to the swap file and the other key of the pair is used to decrypt the page when moving the page from the swap file to the volatile memory.

20 (Canceled).

21. (Currently amended) The system of ~~claim 18~~claim 17, further comprising:

means for loading the codes into a non-swappable region of the volatile memory that is not moved to the swap file.

22. A system for encrypting files, comprising:

a non-volatile storage device, wherein the non-volatile storage device includes a computer file system, wherein files in the file system are associated with groups.

means for providing, for each group, a group identifier, a list of user identifiers of users allowed to access files in the group, and a first encryption code;

means for receiving a second encryption code for one user identifier;

means for receiving an input/output (I/O) request from a requesting user identifier with respect to a target file, wherein one second encryption code has been received for the user identifier;

means for determining the group associated with the target file and the first encryption code for the group;

means for using the determined first encryption code to encrypt the target file to write the target file to the non-volatile storage device if the I/O request is a write operation; and

means for performing if the I/O request is a read operation to read the target file from the non-volatile storage device:

(i) determining whether the requesting user identifier is in the list for the determined group; and

(ii) if the requesting user identifier is in the list, then using the second encryption code for the user identifier to decrypt the target file.

23. (Original) The system of claim 22, further comprising:

means for generating, for each group, a public and private encryption key pair using a public key encryption algorithm, wherein the first encryption code for the group is one of the generated public key or private key and the second encryption code is the other one of the public or private key generated for the group.

24. (Original) The system of claim 23, further comprising:

means for receiving a plurality of keys from the user, wherein each received key is used to decrypt files associated with one group identifier.

25. (Original) The system of claim 23, further comprising:

means for generating an index entry in a non-swappable region in the volatile memory; and

means for adding to the index entry the user identifier of the user that provided the key, the group identifier associated with the received key, and the received key.

26. (Original) The system of claim 25, wherein the index entry for the user identifier and group identifier is only generated if the user identifier is included in the list associated with the group identifier, and wherein the user identifier cannot perform a read access for the target file if there is no index entry for the group identifier associated with the target file and the user identifier.

27. (Original) The system of claim 25, wherein files read and decrypted from the non-volatile storage device are cached in the volatile memory, further comprising:

returning the unencrypted file from the cache to the requesting user identifier if the requested file is unencrypted in the cache and if the requesting user identifier is in the list associated with the group identifier and if there is one index entry for the user identifier and group identifier in the volatile memory.

28. (Original) The system of claim 22, wherein the second encryption code is accessed from a removable storage medium.

29. (Original) A system for encrypting files, comprising:

a volatile memory;

a non-volatile storage device, wherein data is capable of being transferred between the volatile memory and non-volatile storage device;

means for generating an encryption code to encrypt a file and a decryption code to decrypt one file encrypted with the encryption code;

means for loading the decryption code into the volatile memory, wherein the decryption code is erased from the volatile memory when the computer reboots;

means for encrypting files with the encryption code to transfer from the volatile memory to the non-volatile storage device; and

means for decrypting files with the decryption code maintained in the volatile memory to transfer from the non-volatile storage device to the volatile memory.

30. (Original) The system of claim 29, further comprising:

means for generating a new encryption and decryption codes when the computer reboots, wherein the new encryption code is used to transfer files from the volatile memory to the non-volatile storage device and wherein the new decryption code is used to transfer files from the non-volatile storage device to the volatile memory as part of a read operation.

31. (Original) The system of claim 29, wherein the decryption code is loaded into a non-swappable region of the volatile memory.

32. (Original) The system of claim 29, wherein the files are transferred between the volatile memory and non-volatile storage as part of a virtual memory paging operation.

33. (Currently amended) An article of manufacture including program logic for encrypting data in a computer in communication with a volatile memory and non-volatile storage device, by:

initiating a paging operation to move ~~encrypting~~ pages in the volatile memory ~~to move~~ to a swap file in the non-volatile storage device, wherein the non-volatile storage devise is ~~as~~ part of a virtual addressing system;

generating codes to use to encrypt and decrypt the pages, wherein the codes are permanently lost if the computer performs a boot operation;

encrypting the pages in the volatile memory;

moving the encrypted pages from the volatile memory to the swap file; and

upon receiving a subsequent request to transfer the encryhpted pages from the swap file to the volatile memory,

decrypting the encrypted pages in the swap file ~~to move back into the volatile memory~~; and moving the decrypted pages ~~in the swap file~~ back into the volatile memory.

34 (Canceled).

35. (Currently amended) The article of manufacture of ~~claim 34~~claim 33, wherein the codes comprise a public/private key pair generated using a public key cryptography algorithm, wherein one key of the pair is used to encrypt the pages moved to the swap file and the other key of the pair is used to decrypt the page when moving the page from the swap file to the volatile memory.

36 (Canceled).

37. (Currently amended) The article of manufacture of ~~claim 34~~claim 33, wherein the codes are loaded into a non-swappable region of the volatile memory that is not moved to the swap file.

38. (Original) An article of manufacture including program logic for encrypting files in a computer file system in communication with a volatile memory and a non-volatile storage device, wherein files in the file system are associated with groups by:

providing, for each group, a group identifier, a list of user identifiers of users allowed to access files in the group, and a first encryption code;

receiving a second encryption code for one user identifier;

receiving an input/output (I/O) request from a requesting user identifier with respect to a target file, wherein one second encryption code has been received for the user identifier;

determining the group associated with the target file and the first encryption code for the group;

if the I/O request is a write operation, then using the determined first encryption code to encrypt the target file to write the target file to the non-volatile storage device; and

if the I/O request is a read operation to read the target file from the non-volatile storage device, then performing:

(i) determining whether the requesting user identifier is in the list for the determined group; and

(ii) if the requesting user identifier is in the list, then using the second encryption code for the user identifier to decrypt the target file.

39. (Original) The article of manufacture of claim 38, further comprising:

for each group, generating a public and private encryption key pair using a public key encryption algorithm, wherein the first encryption code for the group is one of the generated public key or private key and the second encryption code is the other one of the public or private key generated for the group.

40. (Original) The article of manufacture of claim 39, further comprising receiving a plurality of keys from the user, wherein each received key is used to decrypt files associated with one group identifier.

41. (Original) The article of manufacture of claim 39, further comprising:

generating an index entry in a non-swappable region in the volatile memory; and

adding to the index entry the user identifier of the user that provided the key, the group identifier associated with the received key, and the received key.

42. (Original) The article of manufacture of claim 41, wherein the index entry for the user identifier and group identifier is only generated if the user identifier is included in the list associated with the group identifier, and wherein the user identifier cannot perform a read access for the target file if there is no index entry for the group identifier associated with the target file and the user identifier.

43. (Original) The article of manufacture of claim 41, wherein files read and decrypted from the non-volatile storage device are cached in the volatile memory, and wherein if the requested file is unencrypted in the cache, returning the unencrypted file from the cache to the requesting user identifier if the requesting user identifier is in the list associated with the group identifier and there is one index entry for the user identifier and group identifier in the volatile memory.

44. (Original) The article of manufacture of claim 38, wherein the second encryption code is accessed from a removable storage medium.

45. (Original) An article of manufacture including program logic for encrypting files in a computer in communication with a volatile memory and non-volatile storage device by:

generating an encryption code to encrypt a file and a decryption code to decrypt one file encrypted with the encryption code;

loading the decryption code into the volatile memory, wherein the decryption code is erased from the volatile memory when the computer reboots;

encrypting files with the encryption code to transfer from the volatile memory to the non-volatile storage device; and decrypting files with the decryption code maintained in the volatile memory to transfer from the non-volatile storage device to the volatile memory.

46. (Original) The article of manufacture of claim 45, further comprising: generating a new encryption and decryption codes when the computer reboots, wherein the new encryption code is used to transfer files from the volatile memory to the non-volatile storage device and wherein the new decryption code is used to transfer files from the non-volatile storage device to the volatile memory as part of a read operation.

47. (Original) The article of manufacture of claim 45, wherein the decryption code is loaded into a non-swappable region of the volatile memory.

48. (Original) The article of manufacture of claim 45, wherein the files are transferred between the volatile memory and non-volatile storage as part of a virtual memory paging operation.